

**AMENDMENTS TO THE CLAIMS**

Please amend the claims as follows.

1. (Currently Amended) A method for tracing an instrumented program on a system ~~during booting~~, comprising:  
    booting the system;  
    loading, after the system has booted, object code defining enabling information into a property file associated with a tracing framework; and  
    rebooting the system after loading the object code,  
    wherein while the system is rebooting:  
        processing the property file is processed to enable the tracing framework, wherein enabling the tracing framework comprises creating an anonymous consumer state<sub>2</sub>[[;]] and  
        ~~tracing~~ the instrumented program is traced using the enabled tracing framework, and  
        wherein the system is unable to execute a user-level program while rebooting.
2. (Currently Amended) The method of claim 1, further comprising:  
    associating the anonymous consumer state with a consumer after the system has completed rebooting.
3. (Original) The method of claim 2, wherein the anonymous consumer state is converted to a consumer state after the anonymous consumer state is associated with the consumer.
4. (Original) The method of claim 3, wherein the consumer can access the information obtained during tracing associated with the anonymous consumer state, after the anonymous consumer state is associated with the consumer.

5. (Original) The method of claim 1, further comprising:  
loading a kernel into the system, wherein the kernel is configured to load the property file as soon as possible after the kernel is loaded.
6. (Original) The method of claim 1, further comprising:  
defining a tracing operation source code; and  
generating the object code using the tracing operation source code.
7. (Original) The method of claim 1, wherein the enabling information defines a probe to enable and an action to perform when the probe is encountered during tracing of the instrumented program.
8. (Original) The method of claim 1, wherein the property file is associated with a tracing framework driver.
9. (Original) The method of claim 7, wherein the property file is processed when the tracing framework driver file is loaded into the system.
10. (Original) The method of claim 1, wherein information obtained during tracing associated with the anonymous consumer state is stored in a kernel-level buffer.
11. (Currently amended) An apparatus for tracing an instrumented program on a system ~~during booting~~, comprising:  
a processor configured to, while the system is rebooting:  
process a property file to enable a tracing framework, wherein enabling the tracing framework comprises creating an anonymous consumer state, and  
execute [[a]] the tracing framework, wherein the tracing framework is configured to support [[an]] the anonymous consumer state and configured to trace the instrumented program using the anonymous consumer state; and  
a memory configured to store [[a]] the property file after the system has booted, wherein the property file is configured to store an object code defining enabling information to

create the anonymous consumer state, and wherein the system is rebooted after the property file is stored,

wherein the system is unable to execute a user-level program while rebooting.

12. (Currently Amended) The apparatus of claim 11, wherein the processor is further configured to execute a consumer, wherein the consumer is configured to claim the anonymous consumer state after the system has completed rebooting.
13. (Original) The apparatus of claim 12, wherein the anonymous consumer state is converted to a consumer state after the anonymous consumer state is claimed by the consumer.
14. (Original) The apparatus of claim 12, wherein claiming the anonymous consumer state comprises associating the anonymous consumer state with the consumer.
15. (Original) The apparatus of claim 14, wherein the consumer can access the information obtained during tracing associated with the anonymous consumer state, after the anonymous consumer state is associated with the consumer.
16. (Previously Presented) The apparatus of claim 11, wherein the processor is further configured to:
  - execute a tracing framework driver associated with the property file, wherein the tracing framework driver configured to instantiate the tracing framework; and
  - execute a kernel, wherein the kernel is configured to load the tracing framework driver and configured to process the property file to enable to the tracing framework.
17. (Original) The apparatus of claim 13, wherein the kernel is configured to load the property file as soon as possible after the kernel is loaded.
18. (Original) The apparatus of claim 11, wherein the enabling information defines a probe to enable and an action to perform when the probe is encountered during tracing of the instrumented program.

19. (Original) The apparatus of claim 11, wherein the object code is generated using tracing operation source code.
20. (Currently amended) A network system having a plurality of nodes, wherein each of the plurality of nodes comprises a processor and a memory, comprising:
- a tracing framework configured to, while at least one of the plurality of nodes is rebooting:
    - support an anonymous consumer state, wherein the anonymous consumer state is created when the tracing framework is enabled, and
    - ~~and configured to~~ trace the instrumented program using the anonymous consumer state; and
  - a property file configured to store an object code defining enabling information to create the anonymous consumer state after the at least one of the plurality of nodes has booted, wherein the at least one of the plurality of nodes is rebooted after the property file is stored,
  - wherein the at least one of the plurality of nodes is unable to execute a user-level program while rebooting,
  - wherein the tracing framework resides on any one of the plurality of nodes, and
  - wherein the property file resides on any one of the plurality of nodes.
21. (Currently Amended) The network system of claim 20, further comprising:
- a consumer configured to claim the anonymous consumer state after the at least one of the plurality of nodes has completed rebooting,
  - wherein the property file resides on any one of the plurality of nodes.
22. (Original) The network system of claim 20, further comprising:
- a tracing framework driver associated with the property file configured to instantiate the tracing framework; and
  - a kernel configured to load the tracing framework driver and configured to process the property file to enable to the tracing framework.

23. (Currently Amended) A computer system configured to tracing an instrumented program on a system during booting, comprising:

a first processor;

a memory;

a storage device; and

software instructions stored in the memory for enabling the computer system to:

boot the system;

load, after the system has booted, object code defining enabling information into a property

file associated with a tracing framework; and

reboot the system after the object code is loaded;

wherein while the system is rebooting:

process the property file is processed to enable the tracing framework, wherein

enabling the tracing framework comprises creating an anonymous consumer

state; and

trace the instrumented program is traced using the enabled tracing framework, and

wherein the system is unable to execute a user-level program while rebooting.

24. (Currently Amended) The computer system of claim 23, further comprising:

software instructions stored in the memory for enabling the computer system to:

associate the anonymous consumer state with a consumer after the system has completed

rebooting.

25. (Original) The computer system of claim 23, further comprising:

defining a tracing operation source code; and

generating the object code using the tracing operation source code.